What DNA sequence tells us about gene regulation

The PhyloGibbs-MP approach

Rahul Siddharthan

The Institute of Mathematical Sciences, Chennai 600 113
http://www.imsc.res.in/~rsidd/

03/11/2007
Outline

1 Introduction

2 PhyloGibbs and PhyloGibbs-MP
   - Overview
   - Describing binding sites and the DNA “background”
   - The Gibbs sampler
   - Incorporating phylogeny
   - Enhancements in PhyloGibbs-MP
   - Performance

3 Recent research
Basics

- DNA is a double-stranded molecule consisting of base-paired nucleic acids A,C,G,T on sugar-phosphate backbone
- Proteins are polymers of amino acids with complex 3D conformations that play a fundamental role in cellular processes and infrastructure
- All proteins used by an organism are encoded in genes in an organism
- “Standard dogma”: DNA makes RNA makes protein
- DNA $\rightarrow$ RNA = “transcription”, RNA $\rightarrow$ protein = “translation”
Why study gene regulation?

- Each cell contains a complete copy of the genome but expresses only a few proteins.
- Proteins are expressed only when required.
- Development, cell cycle, stress response and other biological phenomena are driven by complex patterns of gene regulation.
- Being able to predict, in silico, how genes are regulated is of huge benefit to biologists.
Mechanisms of gene regulation

The production of RNA and proteins from genes is regulated at various levels:

- **Transcriptional:** DNA-binding proteins (transcription factors) that recruit the RNA polymerase (activators) or block it (repressors)
- **Post-transcriptional:** the mRNA is modified or degraded: μRNA, siRNA
- **Pre-transcriptional:** “tightly packed” regions (“heterochromatin”) are transcriptionally silent
- **Chemical modification (methylation) of DNA:** genes can be permanently turned off
Mechanisms of gene regulation

The production of RNA and proteins from genes is regulated at various levels:

- **Transcriptional:** DNA-binding proteins (transcription factors) that recruit the RNA polymerase (activators) or block it (repressors)
- **Post-transcriptional:** the mRNA is modified or degraded: $\mu$RNA, siRNA
- **Pre-transcriptional:** “tightly packed” regions ("heterochromatin") are transcriptionally silent
- **Chemical modification (methylation) of DNA:** genes can be permanently turned off

Best understood of these

Transcription factors (TFs) and their interaction with DNA
Mechanisms of gene regulation

The production of RNA and proteins from genes is regulated at various levels:

- **Transcriptional**: DNA-binding proteins (transcription factors) that recruit the RNA polymerase (activators) or block it (repressors)
- **Post-transcriptional**: the mRNA is modified or degraded: $\mu$RNA, siRNA
- **Pre-transcriptional**: “tightly packed” regions (“heterochromatin”) are transcriptionally silent
- **Chemical modification (methylation) of DNA**: genes can be permanently turned off

**Best understood of these**

Transcription factors (TFs) and their interaction with DNA

**Two long-standing computational problems**

Predicting binding sites for individual TFs, and regulatory regions (cis-regulatory modules, enhancers) where many TFs bind
Computational tasks in transcriptional regulation

- In higher organisms: predicting \textit{cis}-regulatory modules (enhancers, etc.)

  - Standard approach: predict binding sites based on already-characterised transcription factors, cluster them.
  - Use phylogenetic information if possible.
  - Examples: Stubb, Cis-Analyzer, Cluster-Buster, ...

- Within \textit{cis}-regulatory modules, or in small intergenic regions for lower organisms: predict individual binding sites for transcription factors.
  - If many sites for a factor are already known, look for similar sites, using position weight matrices.
  - Or, predict sites ab-initio: look for short motifs that occur often in the DNA background.
  - Two standard approaches: expectation maximisation on mixture models (MEME) and Gibbs-sampling.
Computational tasks in transcriptional regulation

- In higher organisms: predicting *cis*-regulatory modules (enhancers, etc)
  - Standard approach: predict binding sites based on already-characterised transcription factors, “cluster them”
  - Use phylogenetic information if possible
  - Examples: STUBB, CIS-ANALYST, CLUSTER-BUSTER, ...

Within *cis*-regulatory modules, or in small intergenic regions for lower organisms: predict individual binding sites for transcription factors

- If many sites for a factor are already known, look for similar sites, using position weight matrices
- Or, predict sites ab-initio: look for short motifs that occur often

Two standard approaches: expectation maximisation on mixture models (MEME) and Gibbs-sampling
Computational tasks in transcriptional regulation

- In higher organisms: predicting cis-regulatory modules (enhancers, etc)
  - Standard approach: predict binding sites based on already-characterised transcription factors, “cluster them”
  - Use phylogenetic information if possible
  - Examples: STUBB, Cis-Analyist, Cluster-Buster, ...

- Within cis-regulatory modules, or in small intergenic regions for lower organisms: predict individual binding sites for transcription factors
Computational tasks in transcriptional regulation

- In higher organisms: predicting *cis*-regulatory modules (enhancers, etc)
  - Standard approach: predict binding sites based on already-characterised transcription factors, “cluster them”
  - Use phylogenetic information if possible
  - Examples: STUBB, CIS-ANALYST, CLUSTER-BUSTER, ...

- Within *cis*-regulatory modules, or in small intergenic regions for lower organisms: predict individual binding sites for transcription factors
  - If many sites for a factor are already known, look for “similar” sites, using “position weight matrices”
  - Or, predict sites “ab-initio”: look for short “motifs” that occur often in the DNA “background”
  - Two standard approaches: expectation maximisation on mixture models (MEME) and Gibbs-sampling
Overview of PhyloGibbs and PhyloGibbs-MP

PhyloGibbs
http://www.imsc.res.in/~rsidd/phylogibbs/

- A Gibbs-sampling motif-finder
- (Optionally) uses orthologous sequence from closely-related species
- Scores phylogenetically-related regions differently from independently-evolved sequence
- Unlike “footprinting” methods, searches in both conserved and unconserved regions
- Uses a two-stage strategy that finds a best set of predictions by simulated annealing and then statistically evaluates their significance by sampling
- Sensitivity and specificity of predictions substantially better than earlier algorithms
Overview of PhyloGibbs and PhyloGibbs-MP

**PhyloGibbs-MP**

RS, 2007, manuscript in preparation

- Algorithmic improvements, including substantial ($\approx 6\times$) speed improvements via importance sampling scheme
- Allows restricting predictions to small regions in a larger input sequence, thus simultaneously predicting cis-regulatory modules (CRMs) and binding sites
- Can predict CRMs and binding sites *ab initio* or with already-characterized TFs as “informative priors”
- Can perform “differential” motif-finding: find motifs that appear in one set of genes but not in another
- Can output GBROWSE-compatible annotations for easier visualisation of results
Transcription factors bind to fuzzy “motifs” that they recognise in DNA. Shown is a logo for CRP binding sites in *E. coli*. Bottom is a sequence logo for the CAP (catabolite activator protein) family itself.

Position weight matrices

<table>
<thead>
<tr>
<th>NA</th>
<th>kruppel</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>01</td>
<td>10</td>
</tr>
<tr>
<td>02</td>
<td>19</td>
</tr>
<tr>
<td>03</td>
<td>17</td>
</tr>
<tr>
<td>04</td>
<td>12</td>
</tr>
<tr>
<td>05</td>
<td>0</td>
</tr>
<tr>
<td>06</td>
<td>4</td>
</tr>
<tr>
<td>07</td>
<td>3</td>
</tr>
<tr>
<td>08</td>
<td>0</td>
</tr>
<tr>
<td>09</td>
<td>0</td>
</tr>
</tbody>
</table>

//
The normalised weight matrix gives (in theory) the probability, at each position, of seeing any given base at that position, given that the sequence is a binding site for the transcription factor being described.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>G</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0</td>
<td>0.50</td>
<td>0.15</td>
<td>0.35</td>
<td>0.00</td>
</tr>
<tr>
<td>01</td>
<td>0.95</td>
<td>0.05</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>02</td>
<td>0.85</td>
<td>0.10</td>
<td>0.00</td>
<td>0.05</td>
</tr>
<tr>
<td>03</td>
<td>0.60</td>
<td>0.30</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>04</td>
<td>0.00</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
</tr>
<tr>
<td>05</td>
<td>0.20</td>
<td>0.00</td>
<td>0.80</td>
<td>0.00</td>
</tr>
<tr>
<td>06</td>
<td>0.15</td>
<td>0.10</td>
<td>0.75</td>
<td>0.00</td>
</tr>
<tr>
<td>07</td>
<td>0.00</td>
<td>0.00</td>
<td>0.10</td>
<td>0.90</td>
</tr>
<tr>
<td>08</td>
<td>0.00</td>
<td>0.10</td>
<td>0.00</td>
<td>0.90</td>
</tr>
</tbody>
</table>
Position weight matrices (normalised)

If we take the weight matrix for the “hunchback” factor and “sweep” it along the first 4000 bp upstream of the “eve” gene in *Drosophila melanogaster* (normalising by a background probability of 0.25) ...
Position weight matrices (normalised)

...we get
(Digression: Module prediction)

(A) High stringency matches

(B) High stringency matches and clustering filter

(C) Expanded view of even-skipped region

Basic approach: predict binding sites of already-characterised transcription factors, and see where they “cluster” together.

From Berman et al., PNAS 99, 757 (2002)

More sophisticated approaches are possible: eg, Stubb, PhyloGibbs-MP.

Rahul Siddharthan
Describing the DNA “background”

Binding sites are thus represented by “position weight matrices”. The rest of the DNA is assumed to be random. It is known that dinucleotide correlations are significant, so we assume a Markov model for the DNA:

\[ P(b_i) = P(b_i | b_{i-1}, b_{i-2}, \ldots) \]

Model for regulatory regions of DNA:

- Some (unknown) sites evolved to bind proteins and represented as samples from a position weight matrix
- Rest of sequence modelled as a Markov chain with conditional probabilities drawn from actual DNA.
- If we know the weight matrices we can use them to predict likely binding sites.
- In ab initio prediction, we need to predict sites described by unknown weight matrices.
The Gibbs sampler

- An example of Markov-chain Monte-Carlo sampling (moveset differs from Metropolis)
- Other implementations have been made; a popular one is AlignAce (Roth et al., 1998)
- Here I discuss the approach from the point of view of PhyloGibbs
Basics of the Gibbs sampler

Given a sequence of DNA, the hypothesis is that some sites are binding sites for motifs, others are background.
A “configuration” $C$ is a particular selection of motif sites, given by the red “windows”. Given a weight matrix and a background model, the probability of observing the sequence $S$ given a configuration $C$ is

$$P(S|C) = P(\text{windows from WM}) \times P(\text{rest from background}).$$
Basics of the Gibbs sampler

We want $P(C|S)$, which is (by Bayes’ theorem)

$$P(C|S) = \frac{P(S|C)P(C)}{\sum_{C'} P(S|C')P(C')}$$

which, if the prior probability $P(C)$ is constant, is proportional to $P(S|C)$.
The probability that *all* the windows came from the *same* WM $w$ is

$$P(C|w) = \prod_{i=1}^{N} \prod_{n=1}^{L} w_{\alpha_i, n}$$

(with $N$ windows of width $L$) This is done column by column, so for the underlined bases we have $w_{C} w_{G}^{2}$. 
Basics of the Gibbs sampler

For the next column, we have \( w_T^3 \). And so on.
For each background base \( \alpha \), we have a background probability \( b_\alpha \).
It’s convenient to divide by prob that entire sequence came from BG: this gives

\[
P(C|w, b) = \frac{P(\text{windows from WM } w)}{P(\text{windows from BG})} = \frac{w_C w_G^2}{b_C b_G^2} \times \frac{w_T^3}{b_T^3} \times \ldots
\]

(overall normalisation won’t matter)
Basics of the Gibbs sampler

We don’t know the weight matrix $w$, so we integrate $P(C|w)$ over the space of all weight matrices $w$. This can be done exactly:

$$
\int_w \prod_{\alpha} w_{\alpha}^{n_{\alpha}} = \frac{3! \prod_{\alpha} n_{\alpha}!}{(N + 3)!}
$$
Basics of the Gibbs sampler

So we have a well-defined “score” $P(C)$ for every “configuration” $C$.

Gibbs sampler strategy: Sample the space of configurations $\{C\}$ for this score.

If we expect three binding sites:
start with three randomly placed windows,
then pick one window, sample all possible positions, and move to a new position.
Basics of the Gibbs sampler

TTATACCAGTACTCTCTTTGTAGCTTTGTAGGTTTTGTTAAATTAGCGTTG

CGTTGTTTTTTACTATGCGTTTGCTGGCCTAACGTCACAAAATCACTTTT

CAAACGGCGCGTACACTCACGGCGTTAAGTATATCAAACTCCGTCACA

If repeated for a long time, and moveset satisfies detailed balance and is ergodic, each point in “configuration space” should be visited with a frequency proportional to its score.
We can sample for multiple different kinds of binding sites simultaneously, by assigning them different “colours”.

TTATACCAGTACTCTCTTGTAGCTTGTAGAATTTGTAAATTAGCGTTG
CGTTGTTTTTACTATGCGTTTGCTGGCCTAACGTCACAAAATCACTTT
CAAACGGCGCGTACACTCACGGCGTTAAGTATATCAAACTCCGTCACA
Issues

- The moveset has different kinds of moves, all of which must satisfy detailed balance, while the overall moveset must be ergodic.
- Many moves satisfy that, but good convergence is needed.
- How does one introduce flexible, rather than rigid, priors on how many motifs (“colours”)/sites per colour may exist?
  - By default the maximum number of colours and the total number of sites is fixed but not the number of sites per colour.
  - We have moves that can add windows in new or existing colours, or remove existing windows.
  - We can specify the “expected” number of sites, and then use an appropriate chemical potential to limit the number of sites to approximately this, but flexibly.
- We need not just to find the best (maximum a posteriori probability) answer, but to assess its significance.
Tracking: significance assessment in PhyloGibbs

- Set up “labelled list” of windows to track (or several “labelled lists”). For every window $w$ in the system, and for every labelled list $A$, set up a “tracking counter” $N(w, A)$ to see how often it’s co-clustered with the list.

- At each time step, for each labelled list $A$, associate one of the current colours with that list, $C(A)$. This is that colour which presently has the greatest presence of windows from that labelled list.

- For every window in that colour, update its tracking counter: do $N(w, A) \rightarrow N(w, A) + 1$ for each $w \in C(A)$.

- Finally, divide each $N(w, A)$ by number of timesteps, and for every label $A$, sort all windows in order of the tracking score. This gives a measure of significance for how often each window was co-clustered with that labelled list.
Incorporating phylogeny

ACGAGCataCAGTAGCA–AGCAC
ATGAGCacaCAGTCGATACCTC
CCGATCGgt−−−−−ataCAGAC

- Use a multiple sequence alignment program to identify conserved blocks. (Any program with multi-fasta output will do. I’ve written one called “Sigma”.) Assume that all bases inside conserved blocks arose from an ancestral sequence and are not independent.

- “Parse” the sequence into a set of legitimate “windows”, which can extend multi-sequence if in conserved blocks.

- Modify scoring for multi-sequence windows; score single-sequence windows the same way as earlier.

- Sample as before.
Incorporating phylogeny

Use a multiple sequence alignment program to identify conserved blocks. (Any program with multi-fasta output will do. I’ve written one called “Sigma”.) Assume that all bases inside conserved blocks arose from an ancestral sequence and are not independent.

“Parse” the sequence into a set of legitimate “windows”, which can extend multi-sequence if in conserved blocks.

Modify scoring for multi-sequence windows; score single-sequence windows the same way as earlier.

Sample as before.
Modified scoring for aligned windows

Assume “star topology”: all sequences descended independently from one common ancestor (generalisations possible)

Assume mutation rate $m_i$ for species $i$, and a divergence time $t$, probability of base being conserved from ancestor (“proximity”) is $e^{-m_i t} = q_i$, probability it’s mutated is $1 - q_i$.

We define a “transition probability” $T(\alpha_i|a)$ that an ancestor $a$ evolved into a base $\alpha_i$:

$$T(\alpha_i|a, q_i) = [\delta_{a\alpha_i} q_i + (1 - q_i) w_{\alpha_i}]$$
Modified scoring for aligned windows

ACGAGC atagACAGTAGCA−AGC A C
ATGAGCacagtacGTCGCATACCTC
CCGATCggt-----atagATACGAC

- “Transition probability” \( T(\alpha_i|a, q_i) = [\delta_{a\alpha_i}q_i + (1 - q_i)w_{\alpha_i}] \)
- Note that it has appropriate limits and the right multiplication rule:

\[
\sum_b T(\alpha_i|b, q_1)T(b|a, q_2) = T(\alpha_i|a, q_1q_2)
\]
Modified scoring for aligned windows

ACGAGC\text{tag}ACAGTAGCA-AGCAC
ATGAGCacagtacGTCGCATACCTC
CCGATCggt------atagATACGAC

- "Transition probability" $T(\alpha_i|a, q_i) = [\delta a\alpha_i q_i + (1-q_i)w_{\alpha_i}]$
- Then, the probability that all four bases in a column of a window $W$ evolved from a common ancestor $a$ and are represented by a WM $w$ is

$$P(W|w) = \sum_{a=A,C,G,T} w_a \prod_{i=1}^{N} T(\alpha_i|a, q_i)$$
Modified scoring for aligned windows

ACGACatatgACAGTAGCA−AGCAC
ATGAGCacagtacGTCGCATACCTC
CGATCggt−−−−−atagATACGAC

“Transition probability” \( T(\alpha_i|a, q_i) = [\delta_{a\alpha_i}q_i + (1 - q_i)w_{\alpha_i}] \)

Single column score for one window:
\[
P(W|w) = \sum_{a=A,C,G,T} w_a \prod_{i=1}^{N} T(\alpha_i|a, q_i)
\]

Product over columns, product over all windows, integral over \( w \) gives score for total “configuration” of selected windows; replacing \( w \) by BG prob. \( b \), without integral, gives background score (in general, a complicated polynomial integral; we use approximations)
Enhancements in PhyloGibbs-MP

PhyloGibbs-MP does the following:

- Uses an “importance sampling” scheme for a speed boost
  - Each window has a weight, measured by how often it occurs in practice, and its probability of being picked is proportional to that weight
  - In practice, only a fraction of windows tend to be picked at all, so this gives a speed boost of a factor of 5 to 10
Enhancements in PhyloGibbs-MP

PhyloGibbs-MP does the following:

- Predicts sites that appear preferentially in one set of input motifs and not in others ("differential motif finding")
  - User supplies a prior probability $p$ that a motif occurs in only one set of input sequences
  - The Bayesian posterior for each configuration is multiplied by $(1 - p)$ if sites for a motif occur in more than one set of input sequences
  - Sufficiently strong motifs that appear in multiple sets can still be found if $p < 1$
Enhancements in PhyloGibbs-MP

PhyloGibbs-MP does the following:

- Predicts sites that occur only in short subsequences of longer input sequences (meant to model cis-regulatory modules)
  - This configuration is enforced on the initial configuration
  - Moves consist of deselecting and replacing windows. When replacing windows, only windows within limits determined by present configuration may be selected
  - Limits are updated both when removing and when replacing windows.
  - One can also require that each CRM contain not more than \( n \) different kinds of sites
Enhancements in PhyloGibbs-MP

PhyloGibbs-MP does the following:

- Uses WMs for already-characterised TFs as “informative priors” to improve predictions
  - These are used to give each window a prior probability of being a binding site, ie of being selected during the sampling moves
  - These prior probabilities are calculated from binding probabilities of TFs, in a manner similar to STUBB
  - Erik van Nimwegen originally wrote a different implementation (available in PhyloGibbs 1.0 and higher, but not documented in the paper). New version is, I believe, more flexible and less aggressive
Enhancements in PhyloGibbs-MP

PhyloGibbs-MP does the following:

- Produces GBrowse-compatible annotations for easy visualisation of the output
  - Examples coming up...
Performance on yeast

PhyloGibbs was benchmarked on SCPD (Zhang et al), a collection of experimentally-annotated binding sites in yeast from which we removed excessively long or short footprints to obtain sites in 200 promoters. We benchmarked PhyloGibbs-MP on the same dataset.

(compare figure in RS et al., PLoS Comput Biol, 2005)
Differential motif prediction

We take two sets of synthetic sequence, each containing 5 “orthologous” sequences of 600bp each. We embed 5 copies per sequence of a common motif $A$ in both sets, and 5 copies each of two “differential” motifs $B$ and $C$ in one set each, and test the performance.
Differential motif prediction

We took large numbers of pairs of promoter sequences for which different regulatory factors are listed in SCPD, and measured PhyloGibbs-MP’s performance.
Predicting cis-regulatory modules

We used the RedFly database of cis-regulatory modules (Halfon et al.), and selected intergenic sequence of 10000 to 20000 bp that encompassed one or more CRMs. We rejected cases where either there were too many CRMs for a gene, extending the required sequence to beyond 20000 bp, or there was very little intergenic sequence beyond the annotated CRM.

Priors were a set of 75 PWMs extracted from the FlyReg database.
Predicting *cis*-regulatory modules: *eve* enhancers

Performance, even without prior WMs, comparable to dedicated CRM prediction programs like Stubb.
The *duf/kirre* enhancer

Collaboration with K G Guruharsha and K VijayRaghavan (NCBS, Bangalore)

- *duf* (aka *kirre*) plays a key role in selection of founder cells in muscle formation in *D. melanogaster*
- Regulation of this gene is of great interest
- Deletion constructs of *duf* promoter by K. G. Guruharsha suggest a complex spatiotemporal pattern of regulation where different pieces of the promoter control expression in different parts of the mesoderm at different stages of development
- We made a detailed study of the regulation of this gene, using Stubb and PhyloGibbs-MP, and using the FlyReg database to construct position weight matrices for several transcription factors of interest
The *duf/kirre* enhancer

Deletion constructs (left), expression (right)

(KG Guruharsha, M Ruiz-Gomez, HA Ranganath, RS, K VijayRaghavan, in preparation)
The *duf/kirre* enhancer

Upstream enhancer structure predicted for the *dumbfounded* or *kirre* gene

(from K G Guruharsha, M Ruiz-Gomez, H A Ranganath, RS, K VijayRaghavan, in preparation)
duf enhancer: detailed view of 11kb upstream

(from K G Guruharsha, M Ruiz-Gomez, H A Ranganath, RS, K VijayRaghavan, in preparation)
Related ongoing research

- with K G Guruharsha, K VijayRaghavan (NCBS) et al.
  Detailed study of regulation of *duf/kirre* in *D. melanogaster* and its role in myogenensis (MS in preparation)
- with K Sanyal (JNCASR) et al.
  Centromere formation in Candida species, and sequence signatures (nearing completion)
- with B Saha (NCCS), P Sadhale (IISc), G Menon, S Sinha (IMSc)
  CD40 signalling in uninfected and *Leishmania*-infected macrophages (ongoing work)
- with S Pyne (Broad Institute/MIT)
  Gene regulation in recently sequenced fission yeast species (discussion stage)
- with S Acharya (IIT-KGP; IASc/INSA summer fellow)
  A simple visualisation program for genomic features and annotations (in progress, but already useful)
- Sigma: a multiple sequence alignment program for non-coding DNA (R.S., BMC Bioinformatics 2006, enhancements in progress)
Future work

- PhyloGibbs-MP already has novel capabilities as a motif finder (CRM prediction, differential motif finding) and superior performance on real data. But improvements are still possible.

- Improve the background model. A Markovian background model misses a lot; my current research explores how to go beyond this. It appears that, with a better background model, functional regions like CRMs “stand out” from the background.

- Some long-term goals are to integrate multiple-sequence-alignment, and microarray data analysis, into PhyloGibbs-MP.

- For better performance on large data sets, a parallel (MPI) version is planned for use on clusters.
Acknowledgements

- Erik van Nimwegen (Basel), Eric Siggia (Rockefeller, NY) – collaborators on PhyloGibbs
- K VijayRaghavan, K G Guruharsha (NCBS, Bangalore), Bhaskar Saha (NCCS, Pune), Parag Sadhale (IISc, Bangalore), Kaustuv Sanyal (JNCASR, Bangalore), Gautam Menon, Sitabhra Sinha, Vani Vemparala (IMSc, Chennai) – collaborations and discussions
- Various users of PhyloGibbs and Sigma for their feedback and comments

Thanks all!